Teamacaron
取扱説明書

1.A0.00
2010年4月

目次

【修正履歴】

Version 1.A0.00

・新規

1. はじめに

1.1 製品構成

Teamacaron 開発キットには以下の内容が含まれています。

○ Teamacaron ハードウェア

詳細は「Teamacaron ハードウェア仕様書」を参照してください。

○ 製品マニュアル

Teamacaron取扱説明書(PDF)
本製品の内容および操作方法に関する説明書です。

PMC T-Kernel Extension 説明書(PDF)
本製品に含まれる T-Kernel Extension に関する説明書です。

PMC T-Shell 説明書 (HTML形式)
PMC T-Shell の API 説明書です。

PMC T-Shell プログラミング解説書 (PDF形式)
PMC T-Shell のプログラミング解説書です。

ライブラリ説明書(PDF)
本製品に含まれるC言語ライブラリに関する説明書です。

デバイスドライバ説明書(PDF)
本製品に含まれるデバイスドライバに関する説明書です。

USBマネージャ取扱説明書(PDF)
本製品に含まれる USBマネージャに関する説明書です。

開発ツール説明書(PDF)
本製品に含まれる T-Kernel および T-Kernel Extension 上で動作する開発ツールの説明書です。

Teamacaron GNU開発環境(Linux版)説明書(PDF)
本製品に含まれる Linux 環境で動作する GNU 開発環境のインストール方法および操作方法に関する説明書です。

Teamacaron GNU開発環境(Windows版)説明書(PDF)
本製品に含まれる Windows上の Cygwin 環境で動作する GNU 開発環境のインストール方法および操作方法に関する説明書です。

Cygwin インストール方法説明書(PDF)
本製品に含まれる Cygwin 環境上で動作するプログラムを使用する場合に必要になる Cygwin のインストール方法に関する説明書です。

Cygwin サポートツールインストール方法説明書(PDF)

本製品に含まれる Teamacaron GNU開発環境(Windows版)を使用する場合に必要になる Cygwin 用プログラムのインストール方法に関する説明書です。

Cygwin 用 Teamacaron 開発環境インストール方法説明書(PDF)
本製品に含まれる Teamacaron GNU開発環境(Windows版)のインストール方法に関する説明書です。

Eclipse インストール方法説明書(PDF)
本製品に含まれる GNU開発環境(Eclipse版)を使用する場合に必要になる Eclipse のインストール方法に関する説明書です。

Eclipse 用 T-Kernel開発環境インストール方法説明書(PDF)
本製品に含まれる GNU開発環境(Eclipse版)のインストール方法に関する説明書です。

GNU開発環境(Eclipse版)説明書(PDF)
本製品に含まれる Windows上の GUI で動作する Eclipse版開発環境の操作方法に関する説明書です。

○ 仕様書

T-Monitor 仕様書 (PDF)
T-Monitor の公式仕様書です。

T-Kernel 仕様書 (PDF)
T-Kernel の公式仕様書です。

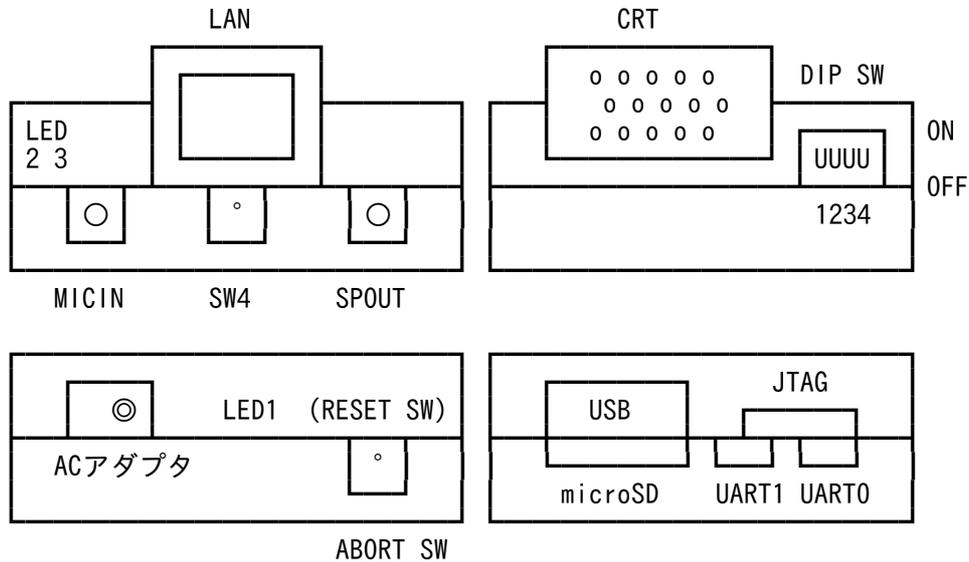
実装仕様書 (PDF)
T-Monitor/T-Kernel そして、デバイスドライバの Teamacaron の実装に依存した部分の詳細な仕様書です。

○ ソフトウェア

ターゲット側
Teamacaron 上で動作する T-Kernel オペレーティングシステムとアプリケーションおよびツールなどの実行プログラムが含まれています。
T-Shell 環境での追加システムが、CD-ROM に入っています。

ホスト側
本製品で使用する GNU 開発環境と、デバイスドライバやサンプルアプリケーションなどのソースが含まれています。開発用ホストマシンとして Linux または、Windows 上の Cygwin 環境を使用します。

1.2 ハードウェア取り扱い方法

Teamacaronのケースの側面には、下図に示す各種のコネクタが付いています。ここでは、使用する上で必要となる事項を簡単に説明します。詳細は「Teamacaron ハードウェア仕様書」を参照してください。

○ DIP SW

4つの DIPSW の設定内容は、以下の通りです。設定は必ず電源 OFF の状態で行ってください。
DIPSW は上側に上げると ON になります。出荷時はすべて OFF(下側)になっています。

		OFF	ON
SW-1	起動選択	自動ブート	モニタ起動
SW-2	USB ストレージ機能	無効	有効
SW-3	(未使用)		
SW-4	(未使用)		

○ LED1,2,3

電源 ON のとき、緑色の LED1 が点灯します。さらに、赤色の LED が 2個あり、T-Monitor のサービスコールを使用して、プログラムで点灯させることができます。

○ microSD スロット

システムディスク用の microSD カードスロットです。

○ ABORT SW(SW3)

割込みを発生させて、T-Monitor に制御を移行します。T-Monitor のコマンドによりシステムを再開出来ます。プログラムのデバッグ用に使用します。

○ RESET SW

システムをリセットして再起動します。通常は使用する必要はありませんが、リセットが必要になったときは、このボタンを押してください。

○ AC アダプタ

開発キット付属の AC アダプタ(5V)を接続します。接続すると自動的に電源が入りシステムが起動します。

○ UART(シリアル)0/1 コネクタ

UART0/1 ともに 3 ピンのヘッドコネクタで制御信号線はありません。
UART0 は、デバッグコンソール接続用のコネクタです。
UART1 は、プログラムで自由に利用することができます。

○ CRT コネクタ

CRT/LCD などのディスプレイを接続する一般的なアナログRGB コネクタです。
画面の解像度は、VGA(640 x 480) か SVGA(800 x 600)を選択できます。標準では VGA (640 x 480)に選択されています。

○ USB コネクタ

USB ホストのコネクタです。USB 2.0 に対応しています。

○ LAN コネクタ

10 Base-T、または 100 Base-TX の LAN が接続できます。

2. インストール方法

Teamacaron には ROM が搭載されていません。そのため、本体のみで起動することは出来ません。起動させるためには microSD カードにソフトウエアをインストールする必要があります。

起動ディスクを作成するには、Windows PC と microSD(SDカード)(※)、そして、microSD を読み書きできる環境が必要です。これらを用意してください。

※Teamacaron は microSDHCカードからは起動出来ません。

2.1 Windows での起動ディスク作成

(1) microSD の準備

Windows PC を使って空の microSD カード(FAT フォーマット(※))に、Windows 上で下記の３つのファイルをコピーします。

```
jp/soft/sdboot.bin
jp/soft/tmonitor.bin
jp/soft/romdisk.bin
```

※FAT32 は使用出来ません。必ず FAT(FAT16)フォーマットしてください。

(2) デバッグコンソールの準備

UART0 コネクタに開発キット付属の Teamacaron 用シリアル変換ケーブルを介して RS-232C クロスケーブルを接続して、他方を開発用パソコンに接続し、パソコン上で通信ソフトウエアを起動してください。

デバッグコンソールの通信仕様は以下の通りです。

通信速度	115,200 bps
データ長	8 ビット
ストップビット	1 ビット
パリティ	なし
フロー制御	XON/XOFF (ハードフロー制御(RS/CS)は利用不可)
文字コード	ASCII / EUC コード
受信行末	CR (0x0d)
送信行末	CRLF (0x0d, 0x0a)

(3) microSD カードを Teamacaron の microSD スロットに装着して起動します。

正常に起動しますとデバッグコンソールにプロンプト [/SYS]% が表示されて入力待ちになります。

(4) デバッグコンソールから hdpart で区画設定をします。

先頭区画を 32MB以上とし、残りを２番目の区画とします。そして２番目の区画を起動区画とします。

以下に作業例を示します。

```
[/SYS]% hdpart pcb
pcb   [C:955  H:64  S:63  B:3862528 (1886 MB)]
No System    Boot   StartCHS      EndCHS        SecNo     SecCnt     Size
1  06 DOS     00     0:  2:10   956: 63:63        135    3858489  1884 MB
2  00 ------  00     0:  0: 0     0:  0: 0          0          0     0 KB
3  00 ------  00     0:  0: 0     0:  0: 0          0          0     0 KB
4  00 ------  00     0:  0: 0     0:  0: 0          0          0     0 KB
** Create/Delete/Boot/Edit/Quit ? d
Delete PartNo (1-4,All) ? 1
No System    Boot   StartCHS      EndCHS        SecNo     SecCnt     Size
1  00 ------  00     0:  0: 0     0:  0: 0          0          0     0 KB
2  00 ------  00     0:  0: 0     0:  0: 0          0          0     0 KB
3  00 ------  00     0:  0: 0     0:  0: 0          0          0     0 KB
4  00 ------  00     0:  0: 0     0:  0: 0          0          0     0 KB
** Create/Delete/Boot/Edit/Update/Quit ? c
Create PartNo (1-4) ? 1
Size [GB/MB/KB,All] (<1886MB) ? 32MB
No System    Boot   StartCHS      EndCHS        SecNo     SecCnt     Size
1  13 BTRON   00     0:  1: 1    16: 63:63         63      68481    33 MB
2  00 ------  00     0:  0: 0     0:  0: 0          0          0     0 KB
3  00 ------  00     0:  0: 0     0:  0: 0          0          0     0 KB
4  00 ------  00     0:  0: 0     0:  0: 0          0          0     0 KB
** Create/Delete/Boot/Edit/Update/Quit ? c
Create PartNo (1-4) ? 2
Size [GB/MB/KB,All] (<1853MB) ? a
No System    Boot   StartCHS      EndCHS        SecNo     SecCnt     Size
1  13 BTRON   00     0:  1: 1    16: 63:63         63      68481    33 MB
2  13 BTRON   00    17:  0: 1   957: 61:61      68544    3793984  1852 MB
3  00 ------  00     0:  0: 0     0:  0: 0          0          0     0 KB
4  00 ------  00     0:  0: 0     0:  0: 0          0          0     0 KB
** Create/Delete/Boot/Edit/Update/Quit ? b
Boot PartNo (1-4,Clear) ? 2
No System    Boot   StartCHS      EndCHS        SecNo     SecCnt     Size
1  13 BTRON   00     0:  1: 1    16: 63:63         63      68481    33 MB
2  13 BTRON   80    17:  0: 1   957: 61:61      68544    3793984  1852 MB
3  00 ------  00     0:  0: 0     0:  0: 0          0          0     0 KB
4  00 ------  00     0:  0: 0     0:  0: 0          0          0     0 KB
** Create/Delete/Boot/Edit/Update/Quit ? u
** pcb: Updated Master Boot Block
[/SYS]%
```

※ hdpart コマンドでは、必ず Boot 区画の設定を行ってください。

(5) ２番目の区画を BTRON 形式にフォーマットします。

```
[/SYS]% format -b pcb1 SYSTEM
```

※ format コマンドでは起動用のブートブロックを書き込むために、必ず -b 指定を行ってくだ

さい。

(6) microSD にシステムをコピーします。

```
[/SYS]% att pcb1 A
[/SYS]% rcp -r /SYS /A/=
[/SYS]% det pcb1
```

(7) 一旦終了して電源を切る

```
[/SYS]% exit
[IMS]% exit
```

(8) Teamacaron に装着していた microSD カードを取り出し、Windows PC に装着して、FATフォーマット(※)します。

※ FAT32 は使用出来ません。必ず FAT(FAT16)フォーマットしてください。
※ エクスプローラ上で microSD に対応するドライブをマウス右ボタンクリックして、「フォーマット」を選択してください。「ファイルシステム」が「FAT」であることを確認してください。

(9) microSD カードに、下記のファイルをコピーします

```
jp/soft/sdboot.bin
jp/soft/tmonitor.bin
jp/soft/install.bz
jp/soft/install_tshell.bz
```

※ このとき、romdisk.bin はコピーしません。

(10) Windows PC から microSD カードを取り外し、Teamacaron に装着して Teamacaron の電源を入れ直します。

※ システムが立ち上がり、デバッグコンソールにプロンプト [/SYS]% が表示されて入力待ちになります。

(11) 基本システムをインストールします。

```
[/SYS]% att -m pcb0 M
[/SYS]% expf -U /M/install.bz
[/SYS]% install /SYS
[/SYS]% rm -r install
```

※ install を実行すると「インストールが完了しました。再起動してください。」と表示されますが、そのまま操作を続けてください。

(12) 続いて T-Shell をインストールします。

```
[/SYS]% expf -U /M/install_tshell.bz
[/SYS]% install_tshell /SYS
```

```
[/SYS]% rm -r install_tshell
[/SYS]% det -u pcb0
```

※ install_tshell を実行すると「インストールが完了しました。再起動してください。」と表示されますが、そのまま操作を続けてください。

(13) システムを再起動すると、T-Shell が起動します。

```
[/SYS]% exit
[IMS]% exit -1
```

以上で、インストールが完了しました。

※ コンソールウィンドウやデバッグコンソールから以下のコマンドを実行すると、デバッグモードを切り替えることができます。デバッグモードでない場合は、デバッグコンソールからの操作はできなくなりますが、システム終了により自動的に電源オフするようになります。

```
[/SYS]% debugmode 0     -- デバッグモードとしない
[/SYS]% debugmode 1     -- デバッグモードとする
```

2.2 動作確認

(1) CRT コネクタに VGA(640 x 480)の表示が可能なディスプレイを接続してください。

(2) USB コネクタにマウス(およびキーボード)を接続してください。

※ キーボードは必須ではありません。

(3) LAN を使用する場合は、LAN コネクタに LAN ケーブルを接続してください。

※ LAN は必須ではありません。

(4) デバッグコンソールの準備

UART0 コネクタに開発キット付属の Teamacaron 用シリアル変換ケーブルを介して RS-232C クロスケーブルを接続して、他方を開発用パソコンに接続し、パソコン上で通信ソフトウェアを起動してください。

デバッグコンソールの通信仕様は以下の通りです。

通信速度	115,200 bps
データ長	8 ビット
ストップビット	1 ビット
パリティ	なし
フロー制御	XON/XOFF (ハードフロー制御(RS/CS)は利用不可)
文字コード	ASCII / EUC コード
受信行末	CR (0x0d)
送信行末	CRLF (0x0d, 0x0a)

(5) AC アダプタコネクタに開発キット付属の AC アダプタを接続すると、microSD 上のシステムが起動して、接続したディスプレイ画面上に初期画面が表示されます。

※ デバッグコンソールには、一連の起動メッセージが表示されて、CLI のコマンド入力待ちの状態になります。

(6) 初期画面上で、マウス/キーボードを使用して、サンプルの動作を確認してください。

※ 基本的な操作方法に関しては超漢字V(別売)の取扱説明書を参照してください。
※ かな漢字変換は使用できません。
※ 追加のフォントをインストールしないと表示できない文字があります。フォントの追加は後述の「フォント」を参照してください。
※ 接続した LAN 環境で DHCP サーバが使える場合は、「ブラウザ」を開いて LAN へのアクセスが可能です。DHCP サーバが使えない場合は設定が必要になります。後述の「ネットワーク」を参照してください。

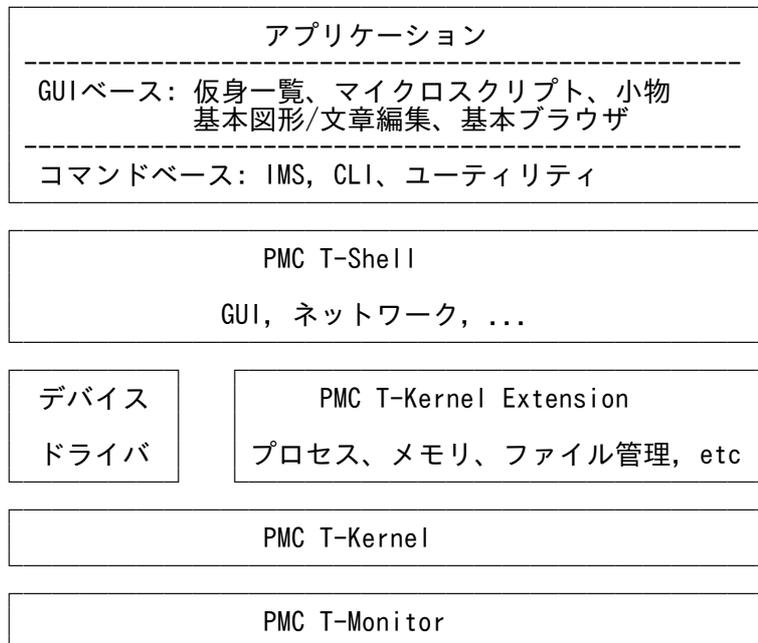
(7) 終了するには、初期ウィンドウを閉じて、背景画面だけが表示された状態としてください。

起動した microSD カードは、デバッグモードになっているため、自動的に電源は OFF しません。以下のようにデバッグコンソールから exit コマンドを 2 回入力することにより、電源が OFF します。

```
[/SYS]% exit
[IMS]% exit
```

3. ソフトウェア構成

3.1 全体構成

本製品の全体のソフトウェア構成を以下に示します。

○ PMC T-Monitor

T-Monitor仕様書に準拠したモニタです。詳細は「T-Monitor 仕様書」および「実装仕様書」を参照してください。

○ PMC T-Kernel

T-Kernel仕様書に準拠したリアルタイムカーネルです。T-kernel/OS, T-kernel/DS, T-kernel/SM を含んでいます。詳細は「T-Kernel 仕様書」および「実装仕様書」を参照してください。

○ PMC T-Kernel Extension

本製品用に、T-Kernel/OS のサブシステム機能を利用して構築した OS の拡張部分で、BTRON3 仕様に準拠した以下の機能を含んでいます。詳細は「PMC T-Kernel Extension 説明書」を参照してください。

プロセス/タスク管理、メッセージ管理、プロセス/タスク間同期通信管理、グローバル名管理、メモリ管理、ファイル管理、イベント管理、デバイス管理、時計管理、システム管理、UNIX(ファイル)エミュレータ

この拡張により、仮想記憶 OS を実現しており、開発環境としてファイルやプロセスを利用できるようになっています。

○ PMC T-Shell

本製品用に、T-Kernel/OS のサブシステム機能を利用して構築したミドルウェア群であり、BTRON3 仕様に準拠した以下の機能を含んでいます。GUI やネットワークを使用したアプリケーションの開発のための多くの機能を提供しています。

ディスプレイプリミティブ、フォントマネージャ
フォントデータ(ドット、TrueType)
ウィンドウマネージャ、メニューマネージャ、パーツマネージャ、パネルマネージャ、トレーマネージャ、データマネージャ
テキスト入力プリミティブ
実身／仮身マネージャ
TCP/IPマネージャ

○ デバイスドライバ

本製品用に、T-Kernel/SM デバイス管理機能に基づいた以下のデバイスドライバが含まれています。デバイスドライバの詳細は「デバイスドライバ説明書」および「実装仕様書」を参照してください。

USB マネージャ (バスドライバ)
時計(RTC)、コンソール(シリアル)、システムディスク(microSD、USBメモリ)、KB/PD(USB KB/MOUSE)、スクリーン、LAN

○ アプリケーション

下記のコマンドベースの開発用ツールが含まれています。

IMS (Initial Monitor System)
　詳細は「開発ツール説明書」の「IMS」の章を参照してください。
CLI (Command Line Interpreter)
　詳細は「開発ツール説明書」の「CLI」の章を参照してください。
フォーマッタ、その他各種ツール、テストプログラムなど
　詳細は「開発ツール説明書」の「ユーティリテイ」の章を参照してください。

下記の GUI アプリケーションが含まれています。

仮身一覧、マイクロスクリプト、基本図形/文章編集、基本ブラウザ、各種小物

詳細は超漢字Ｖ(別売)に含まれる電子マニュアルを参照してください。

3.2 ファイル構成

microSD カード上のディスク（pcb1）には、以下のファイルが格納されています。

ファイル	説明
SBOOT	2次ブートプログラム
KERNEL.SYS	OS 核（ T-Kernel, Extension, ドライバー部）
SYSCONF	システムコンフィグレーションファイル（EUC テキスト）
DEVCONF	デバイスコンフィグレーションファイル（EUC テキスト）
STARTUP.CMD	システム起動コマンド（EUC テキスト）
STARTUP.CLI	CLI 起動コマンド（EUC テキスト）
chgsys	システムディスク切り替え
screen	画面(スクリーン)ドライバ
kbpd	KB/PD ドライバ
lowkbpd	KB/PD 実 I/O ドライバ
rsdrv	RS ドライバ
unixemu	UNIX(ファイル)エミュレータ
startimg	初期画面イメージ
cli	CLI
.xcli	CLI コマンドファイル（EUC テキスト）
bin/	CLI ユーティリティ
hdpart	簡易ディスク区画作成
format	ディスクフォーマット
dd	ディスクダンプツール
ed	簡易行エディタ
expf	解凍ツール
cmp	ファイル比較
devlist	デバイス一覧
mscnv	簡易ファイル変換ツール
usbinf	USB デバイス情報表示ツール
devconf	DEVCONF 変更ツール
sysconf	SYSCONF 変更ツール
debugmode	DEBUGMODE 変更ツール
chver	バージョン変更ツール
chatr	ファイル属性変更ツール
vup	ファイルバージョンアップツール
pacf	アーカイブ作成ツール
fget	ftp によるファイルのダウンロード
fput	ftp によるファイルのアップロード
telnet	簡易 telnet ツール
ping	簡易 ping ツール
netdump	LAN パケット情報の表示(テスト用)
netinf	LAN ドライバ情報の表示(テスト用)
netconf	NETCONF 作成ツール
netstart	ネットワーク起動ツール
ppp	PPP 接続ツール
fbox	$$FONT.BOX ツール
ux/	UNIX(ファイル)エミュレータコマンドツール
pwd	カレントディレクトリのパス名表示
ls	ディレクトリ内容のリスト表示
cp	ファイル(ディレクトリ)のコピー

```
            rm          ファイル(ディレクトリ)の削除
            mv          ファイル(ディレクトリ)の移動、ファイル名変更
            mkdir       ディレクトリの作成
            rmdir       ディレクトリの削除
            chmod       ファイルアクセス権の変更
            fcnv        ファイル変換
netdrv                  LAN ドライバ
font                    フォントマネージャ
font_nottf              フォントマネージャ（TrueType 未対応版）
fpsvr                   フォントサーバー
dp                      ディスプレイプリミティブ
tip                     テキスト入力プリミティブ
hmi                     HMI マネージャ
$$SYSDBOX               システムデータボックス
omgr                    実身/仮身マネージャ
tcpipmgr                TCP/IP マネージャ
sysdmn                  システムデーモンプロセス
logon                   ログオンプロセス
NETCONF                 ネットワークコンフィグレーション情報
$$RELATION.BOX          続柄情報
@USER_ID_CARD           ユーザ情報
$$BGSCREEN.BOX          背景画面
$$CUSTOMIZE.BOX         画面カスタマイズデータ
FONT/                   基本フォント
        基本-明朝固定S
$$FONT.BOX/             追加フォント
        基本-明朝可変
$$PROGRAM.BOX/          プログラム格納場所
        DLED            仮身一覧
        CHGENV          ユーザ環境設定
        SYSENV          システム環境設定
        MSCRIPT         マイクロスクリプト
        CALC            電卓
        CLK             時計
        TXED            基本文章編集
        FGED            基本図形編集
        bbb             基本ブラウザ
        NETENV          ネットワーク設定
        FINDER          仮身参照
        FSEARCH         仮身検索
        NET             仮身ネットワーク
        FORMATTER       ディスク初期化
        MANMOS          原紙箱
        GADGET          小物箱
        DEVICE          ディスク集め
        CONSOLE         コンソール
USR/                    初期ウィンドウ
        サンプル
        原紙箱
        小物箱
```

ブラウザ用紙
キャビネット
原稿用紙
画用紙
スクリプト用紙
lib/ 共有ライブラリ
libapp.so.2
libg.so.2
libstlport.so.2
libux.so.2
libbt.so.2
libdl.so.2
libimg.so.2
libtf.so.2
libwin.so.2

3.3 システムの起動手順

システムの起動は以下の手順で行われます。

(1) CPU のファームウエア が microSD カードの FAT 区画から sdboot.bin をロードし、sdboot に制御を移します。

(2) sdboot が、microSD カードの FAT 区画から T-Monitor(tmonitor.bin) をロードします。

(3) microSD カードの FAT 区画に ROM ディスクイメージ(romdisk.bin)がある場合は、sdboot は ROM ディスクイメージをロードし、ROM ディスクを構成します。

(4) sdboot は ROM 情報を設定した後、T-Monitor に制御を移します。

(5) T-Monitor が、ハードウエアやシステム共有情報などの初期設定を行います。

(6) T-Monitor がディスクの BTRON 区画から PBOOT(１次ブート) をロードし、PBOOT へ制御を移します。

(7) PBOOT がディスクの BTRON 区画から SBOOT(２次ブート) をロードし、SBOOT へ制御を移します。

(8) SBOOT は、ファイルシステム内の以下のファイルをそれぞれメモリに読み込んで、KERNEL.SYS を実行します。

KERNEL.SYS	OS 核 (T-Kernel, Extension, ドライバー部)
SYSCONF	システムコンフィグレーションファイル
DEVCONF	デバイスコンフィグレーションファイル

(9) KERNEL.SYS は、SYSCONF で指定された各種のパラメータにしたがって、初期化を行い、T-Kernel および T-Kernel Extension を動作させます。また、以下のドライバも動作させます。これらのドライバは以降のシステムの立ち上げに必須のため、KERNEL.SYS に含まれています。

USB マネージャ、時計(RTC)、コンソール(シリアル)、システムディスク

(10) OS としてのすべての初期化が終了すると、初期タスクとして IMS が立ち上がり、STARTUP.CMD ファイルを読み込んで、その内容にしたがった処理を行います。

(11) STARTUP.CMD では、他のドライバ、サブシステム、サーバープロセスなどの基本部分を立ち上げたのち、最後に STARTUP.CLI をパラメータとして、CLI を実行します。

(12) CLI は、STARTUP.CLI ファイルの内容を読み込んで、その内容にしたがって処理を行います。

(13) STARTUP.CLI では、/SYS/USR を初期ウィンドウとして、仮身一覧を起動します。

4. システムコンフィグレーション

4.1 GUI 機能

GUI 機能は、大きく 3 つのレベルに分類できます。

インストール直後は、(3)BTRONシステムレベルになっていますが、利用レベルに応じて不要なモジュールを削除して、STARTUP.CMD 内の対応する起動行を削除またはコメントアウトしてください。

†STARTUP.CMD は、Teamacaron上で簡易エディタの ed ツールを使用して直接編集するか、または、開発環境の $BD/kernel/config/em1d512 にインストールされる STARTUP.CMD ファイルをホスト上で編集してダウンロードしてください。
†$BD は開発環境のベースディレクトリを表します。通常、コマンドライン版の開発環境の場合は /usr/local/te になります。Eclipse 版の場合 C:¥eclipse¥plugin¥com.t_engine4u.tl.em1d512.x.y.z_x.y.z¥te になります。

(1) 基本描画レベル

画面描画のための基本機能レベルで、ディスプレイプリミティブにより実現されます。

/SYS/dp	ディスプレイプリミティブ

文字描画を行うためには、フォントマネージャとフォントデータが必要になります。
文字描画を行なわない場合は、削除可能です。

/SYS/font	フォントマネージャ
/SYS/fpsvr	フォントサーバー
/SYS/FONT/*	基本フォントデータ(必須)
/SYS/$$FONT.BOX/*	追加フォントデータ(オプション)

(2) ウィンドウレベル

ウィンドウシステムのレベルで、以下の機能を 1 つにまとめた HMI マネージャにより実現されます。
このレベルの実現のためには、(1) 基本描画レベルは必須となります。

ウィンドウマネージャ
メニューマネージャ
パーツマネージャ
パネルマネージャ
トレーマネージャ
データマネージャ

/SYS/hmi	HMI マネージャ
/SYS/$$SYSDBOX	システムデータボックス
/SYS/tip	テキスト入力プリミティブ

(3) BTRONシステムレベル

実身/仮身を使用した、BTRONシステムレベルは、以下のモジュール/データにより実現されます。このレベルの実現のためには、(1)基本描画レベル、および (2)ウィンドウレベルは必須となります。

/SYS/omgr　実身/仮身マネージャ
/SYS/logon　ログオンプロセス
/SYS/$$RELATION.BOX　続柄情報
/SYS/$$PROGRAM.BOX/*　プログラム格納場所

4.2 ユーザ環境

「ユーザ環境設定」小物で設定した、ユーザの操作環境に関する各種の設定は、以下のファイルに保存されています。

/SYS/@USER_ID_CARD　ユーザID カード

4.3 画面カスタマイズ

画面のカスタマイズに関して、以下のデータファイルが利用できます。

$$BGSCREEN.BOX　背景画面(壁紙)
$$CUSTOMIZE.BOX　画面のカスタマイズ(配色/書体)情報

インストール直後は、サンプルの背景画面が入っています。配色/書体は、標準のみの状態です。

「システム環境設定」小物の「画面-背景/配色/書体」により、背景画面や、画面のカスタマイズ(配色/書体)を行うことができます。

4.4 フォント

フォントの機能は、以下のモジュールにより実現されます。

/SYS/font　フォントマネージャ
/SYS/fpsvr　フォントサーバー
/SYS/FONT/*　基本フォントデータ(必須)
/SYS/$$FONT.BOX/*　追加フォントデータ(オプション)

フォントを利用しない場合は、上記モジュールをすべて削除し、STARTUP.CMD 内の font の起動行を削除またはコメントアウトしてください。

また、ドットフォントのみを使用し、TrueType フォントを使用しない場合は、font_nottf (font よりサイズが小さい) を /SYS/font の代わりに利用することができます。この場合でも fpsvr は必要です。

インストール直後は、以下のフォントデータが登録されています。

/SYS/FONT/基本ー明朝固定S
　12,16ドットの漢字フォント
　8,12,16ドットの英数全角/英数半角/英数比例ピッチフォント

/SYS/$$FONT.BOX/基本-明朝可変

明朝体 TrueType フォント

CD-ROM 内の common¥font ディレクトリには、以下のフォントが含まれています。

† フォントの内容は、超漢字Ｖ(別売)に含まれる「システム環境設定」小物で「書体」の詳細情報を参照してください。Teamacaron上で「システム環境設定」小物では、「書体」の詳細情報は表示できません。

基本-明朝固定S
基本-明朝固定
基本-明朝16
基本英数-明朝固定
補助-明朝固定
補助非漢字-明朝固定
中国-明朝固定
韓国-明朝固定
点字-固定
基本-明朝可変
基本-SS明朝可変
基本-SS細明朝可変
基本-SSゴシック可変
基本-SS教科書可変
基本-SS丸ゴシック可変
基本英数-SSAvalon可変
基本英数-SSCourier可変
基本英数-SSEuromode可変
基本英数-SSHelvetica可変
基本英数-SSHelveticaN可変
基本英数-SSTimes可変
補助-明朝可変
補助-SS明朝可変
補助-SSゴシック可変
日本第3第4-明朝可変
GT-GT書体可変
GT-GT書体追加可変
大漢和-SS細明朝可変
中国-SS明朝可変
中国-SSゴシック可変
中国-SS仿宋可変
中国-SS楷書可変
韓国-SS明朝可変
韓国-SS明朝可変太字
韓国-SSゴシック可変
韓国-SSゴシック可変太字
韓国-SS宮書可変
韓国-SS丸ゴシック可変
台湾-SS明朝可変
台湾-SSゴシック可変
各国-SS明朝可変
各国-SSゴシック可変

シンボル-iモード絵文字可変
シンボル-序数記号可変
シンボル-陰陽五行可変
シンボル-ホツマ文字可変
点字-印刷用可変
点字-表示用可変

フォントの登録方法/削除方法に関しては、CD-ROM の common¥font ディレクトリ内の readme.txt を参照してください。なお、/SYS/FONT に入っている基本フォントは、変更しないでください。

4.5 ネットワーク

ネットワークの機能は、以下のモジュールにより実現されます。

```
/SYS/netdrv            LAN ドライバ
/SYS/tcpipmgr          TCP/IP マネージャ
/SYS/NETCONF           ネットワークコンフィグレーションファイル
```

ネットワークを全く利用しない場合は、上記モジュールをすべて削除し、STARTUP.CMD 内の netdrv、および tcpipmgr の起動行を削除またはコメントアウトしてください。

ネットワークコンフィグレーションファイル(NETCONF)には、ホスト、DNS サーバー、ドメイン名、ゲートウェイなど設定します。「ネットワーク設定」小物で設定しますが、CLI 上で netconf ツールを利用して設定することもできます。

```
[/SYS]% netconf         # NETCONF の表示
hostname   =
host ip    = 0.0.0.0
dns1name   =
dns1 ip    = 0.0.0.0
dns2name   =
dns2 ip    = 0.0.0.0
domain     =
gateway ip = 0.0.0.0
subnetmask = 255.255.255.0

[/SYS]% netconf c       # NETCONF の作成/変更
hostname   =  ? myhost
host ip    = 0.0.0.0 ?  198.162.0.2
dns1name   =  ? dns1
dns1 ip    = 0.0.0.0 ?  198.162.0.254
dns2name   =  ?
dns2 ip    = 0.0.0.0 ?
domain     =  ?         tshell-test
gateway ip = 0.0.0.0 ?
subnetmask = 255.255.255.0 ?
wlan       = none (n/a/i)? n    # n:noe, a:adhoc, i:infra
```

†wlan は、無線LAN の設定ですが、Teamacaronでは、無線LAN は利用できません。
†netconf ツールは、NETCONF ファイルが存在しないときは自動的に生成しますので、設定

がうまくいかないときは、一度 NETCONF ファイルを削除してから設定し直してください。

```
[/SYS]% rm NETCONF
[/SYS]% netcopnf c
```

インストール直後の NETCONF は、空（ホスト IP アドレス = 0.0.0.0）の設定となっていますが、接続したネットワーク上に DHCP サーバーが存在する場合は、DHCP サーバーから IP アドレスを取得するため、そのまま動作させることが可能です。

ネットワーク機能に関しては、以下のユーティリティが用意されています。それぞれ、パラメータなしで実行すると、簡単なヘルプメッセージが表示されます。

ping	簡易 ping ツール
fget	ftp によるファイルのダウンロード
fput	ftp によるファイルのアップロード
telnet	簡易 telnet ツール
ppp	PPP 接続ツール
netstart	ネットワーク起動ツール
netinf	LAN ドライバ情報の表示(テスト用)
netdump	LAN パケット情報の表示(テスト用) netdump ? で、ヘルプメッセージが表示されます。

ネットワーク機能を利用したプログラミングに関しては、「PMC T-Shell 説明書」の「TCP/IPマネージャ」の章を参照してください。BSD ソケットI/F に準拠した API が用意されています。

また、開発環境の $BD/util/tool/src/* にインストールされたネットワーク関連ツールのソースファイルも参考にしてください。

注)他のマシンからの ping への応答について

TCP/IP マネージャがロードされていても、ネットワークを使用するプログラムが一つも存在しない状態では、ネットワークは停止状態になります。このとき、外部からの ping 要求などにも全く反応しませんので、ご注意ください。
ネットワークを動作状態にするには、ネットワークを使用するプログラムを起動するか、または、TCP/IP マネージャに対して明示的に so_start() を発行する必要があります。
このためのユーティリティとして netstart があります。このプログラムを次のように動作させておけば、ネットワークは常に動作状態になり、外部からの ping 要求にも応答するようになります。

```
[/SYS]% netstart &
```

この netstart や so_start() を実行しなくても、通常の手順でプログラムから TCP/IP 機能を使い始めれば、その時点でネットワークは動作状態になります。ネットワークを使用するプログラムが、明示的に so_start()を発行する必要はありません。

5. ソフトウェア開発方法

5.1 開発環境のインストール

5.1.1 Linux 環境で開発を行う場合

GNU 開発環境のインストールやコンパイル方法に関しては、

・Teamacaron GNU開発環境(Linux版)説明書

を参照してください。その後、後述する「T-Shell 開発環境のインストール」を参照してください。

5.1.2 Windows 上のコマンドライン環境で開発を行う場合

GNU 開発環境のインストールやコンパイル方法に関しては、

・Cygwin インストール方法説明書
・Cygwin サポートツールインストール方法説明書
・Cygwin 用 Teamacaron 開発環境インストール方法説明書
・Teamacaron GNU開発環境(Windows版)説明書

を参照してください。その後、後述する「T-Shell 開発環境のインストール」を参照してください。

5.1.3 Windows 上の GUI 環境で開発を行う場合

GNU 開発環境のインストールやコンパイル方法に関しては、

・Cygwin インストール方法説明書
・Eclipse インストール方法説明書
・Eclipse 用 T-Kernel 開発環境インストール方法説明書
・GNU開発環境(Eclipse版)説明書

を参照してください。開発に関する詳細は

・Teamacaron GNU開発環境(Windows版)説明書

を参考にしてください。
その後、後述する「T-Shell 開発環境のインストール」を参照してください。

5.1.4 T-Shell 開発環境のインストール

(1) Linux 環境で開発する場合

T-Shell を使用する上で必要なツール、ライブラリ、サンプルプログラム等を追加インストールするためには、本 CD-ROM 内の、jp¥soft ディレクトリに含まれている以下のファイルを開発環境のディレクトリ($BD)上で展開してください。

te.tshell.em1d512.tar.gz

例 : % cd /usr/local/te

```
% tar zxpf te.tshell.em1d512.tar.gz
```

(2) Windows 上のコマンドライン環境で開発する場合

「(1) Linux 環境で開発する場合」と同じです。「(1) Linux 環境で開発する場合」を参照してください。

(3) Windows 上の GUI 環境で開発する場合

・PMC T-Shell(Eclipse版)インストール説明書

を参照してください。

(4) インストール内容

追加インストールされる内容は以下の通りです。

```
bappl/
        dbox/*                  データボックスの標準インクルードファイル
                                (stddef.d, libapp.i, libapp.d)
        tsh-sample/*            「PMC T-Shell プログラミング解説書」で参照している
                                サンプルプログラム群
lib/
        em1d512/
                libtf.a         多国語(多漢字)処理ライブラリ
                libtf.so        (../em1d512.dl/libtf.so へのシンボリックリンク)
        em1d512.dl/
                libtf.so.2      多国語(多漢字)処理ライブラリ(共有ライブラリ)
                libtf.so        (libtf.so.2 へのシンボリックリンク)
                libtf.a

util/
        tool/src/               ネットワーク関連ツールソースファイル
                netconf.C       NETCONF 作成ツール
                fget.C          ftp によるファイルのダウン/アップロード
                telnet.c        簡易 telnet ツール
                ping.C          簡易 ping ツール
                ppp.C           PPP 接続ツール
                netstart.c      ネットワーク起動ツール
                tcptest.C       TCP テスト用
                udptest.C       UDP テスト用
```

†データボックスコンパイラに関しては、開発環境に含まれるソースファイル ($(BD)/tool/build/databox/src/databox.C) 内の先頭にある説明を参照してください。

5.2 開発対象ソフトウェアの分類

開発対象となるソフトウェアは、大きく以下の 2 つに分類され、開発方法やプログラムのオブジェクト形式などが異なってきます。

Teamacaron では、プロセスベースのソフトウェアの開発が主体になります。

○ T-Kernel ベースのソフトウェア

T-Kernel の機能を使用するデバイスドライバやサブシステムなどのリアルタイムソフトウェアで、常駐メモリ上で動作します。

○ プロセスベースのソフトウェア

T-Knernel Extension、および T-Shell 上で、プロセスとして仮想メモリ上で動作するソフトウェアで、T-Kernel の機能を直接利用することはできません。
一般のアプリケーションソフトウェアや、ライブラリ相当のミドルウェアなどが相当します。

5.3 T-Kernel ベースのソフトウェア

デバイスドライバなどの、T-Kernel ベースのソフトウェアは、MMU を使用した環境で、システムの共有空間に常駐して動作します。

CLI の recv コマンドを使用して、作業用ディスクに一旦ファイルとして格納してから、CLI または IMS の lodspg コマンドでロード & 実行を開始します。lodspg を終了してもメモリ領域を占有したままとなります。プログラムの終了は unlspg コマンドで行い、プログラム終了後にメモリ領域は解放されます。

ロード時には、自動的にリロケーションが行われます。実際にロードしたアドレスは、lodspg コマンドで表示されますが、CLI の ref spg コマンドで確認することができます。

5.4 プロセスベースのソフトウェア

一般アプリケーションとしてのプロセスベースのソフトウェアは、MMU を使用した環境で、ローカル空間にロードされて動作します。

CLI の recv コマンドを使用して、作業用ディスクに一旦ファイルとして格納してから、CLI または IMS の プログラム実行コマンドでロード & 実行を開始します。

プロセスベースのソフトウェアはユーザレベルの保護レベルで動作しますので、T-Kernel の機能を直接利用することはできません。また、I/O 空間を直接アクセスすることはできません。

プロセスベースのソフトウェアでは、共有ライブラリを使用することができます。共有ライブラリ自体は、システム起動ディスクの /SYS/lib/* に含まれています。共有ライブラリを使用したプログラムの開発方法に関しては「TeamacaronGNU開発環境(Linux版)説明書」または「TeamacaronGNU開発環境(Windows版)説明書」を参照してください。

5.5 プログラム開発方法

PMC T-Kernel Extension、T-Shell を利用できるソフトウェアは、基本的にプロセスベースのソフトウェアとなりますので、開発環境のディレクトリ $BD/bappl で作成してください。

実際の開発方法は、「TeamacaronGNU開発環境(Linux版)説明書」または「TeamacaronGNU開発環境(Windows版)説明書」を参照してください。

PMC T-Shell で提供している API に関しては、「PMC T-Shell説明書」および「PMC T-Shell プログラミング解説書」と、開発環境の $BD/bappl/tsh-sample/* にインストールされるサンプルプログラム群を参照してください。

5.6 マイクロスクリプト開発方法

マイクロスクリプトを使用したプログラムは、Teamacaron上で直接作成することもできますが、超漢字Ｖ(別売)上で作成し、作成した結果をTeamacaron上にコピーして実行することもできます。

マイクロスクリプトに関しては、超漢字Ｖ(別売)の電子マニュアル、または書籍を参照してください。

超漢字Ｖ(別売)で作成したマイクロスクリプトの実身は、USB メモリーなどを利用してTeamacaronへコピー(実身複製)してください。
コピーの方法などは、超漢字Ｖ(別売)の説明書をご参照ください。